IN THE CLAIMS:

The current claims follow. For claims not marked as amended in this response, any difference in the claims below and the previous state of the claims is unintentional and in the nature of a typographical error.

1. (Canceled).

2. (Currently Amended) A pipelined microprocessor detecting a first instruction using first base and offset address values to load data from a first memory location that was previously stored to, wherein the first instruction is detected based upon the first base and offset address values and without computing a memory address equaling the first base address value added to the offset address ~~values~~ value in detecting the first instruction.

3. (Previously Presented) A pipelined microprocessor as claimed in claim 2 wherein the pipelined microprocessor detects a second instruction using second base and offset address values to store data into a second memory location that was previously read from, wherein the second instruction is detected based upon the second base and offset address values and without computing a memory address equaling the second base address value added to the offset address values in detecting the second instruction.

4.-5. (Canceled).

6. (Previously Presented) A pipelined microprocessor as claimed in claim 2 wherein the pipelined microprocessor examines base and offset address values used to access memory locations by store instructions that store data into the memory locations, and detects load instructions that load data from memory locations corresponding to base and offset address values identical to the base and offset address values used by the store instructions.

7. (Previously Presented) A pipelined microprocessor as claimed in claim 3 wherein the pipelined microprocessor examines base and offset address values used to access memory locations by load instructions that load data from the memory locations, and detects store instructions that store data into memory locations corresponding to base and offset address values identical to the base and offset address values used by the load instructions.

8. (Previously Presented) A pipelined microprocessor as claimed in claim 6 wherein the pipelined microprocessor detects identical offset address values and identical base address values in at least one register within the pipelined microprocessor.

9. (Previously Presented) A pipelined microprocessor as claimed in claim 7 wherein the pipelined microprocessor detects identical offset address values and identical base address values in at least one register within the pipelined microprocessor.

10. (Previously Presented) A pipelined microprocessor as claimed in claim 6 wherein the pipelined microprocessor comprises:

an instruction decode stage detecting load instructions that load data from memory locations corresponding to offset address values from an identical and base address values identical to offset address values and base address values used by prior store instructions that store data into the memory locations; and

a bypass element sending a bypass signal to an instruction execution stage of the pipelined microprocessor that indicates that a load instruction uses a base address value and an offset address value identical to a base address value and an offset address value used by a prior store instruction.

11. (Previously Presented) A pipelined microprocessor as claimed in claim 7 wherein the pipelined microprocessor comprises:

an instruction decode stage detecting store instructions that store data into memory locations using offset address values and base address values identical to offset address values and base address values used by prior load instructions that load data from memory locations; and

a bypass element sending a bypass signal to an instruction execution stage of the pipelined microprocessor that indicates that a store instruction uses a base address value and an offset address value identical to a base address value and an offset address value used by a prior load instruction.

12. (Currently Amended) A method for operating a pipelined microprocessor, comprising:

detecting, in the pipelined microprocessor, a first instruction using first base and offset address values to load data from a first memory location that was previously stored to, wherein the first instruction is detected based upon the first base and offset address values and without computing a memory address equaling the first base address value added to the offset address ~~values~~ value in detecting the first instruction.

13. (Previously Presented) A method for operating a pipelined microprocessor as claimed in claim 12, further comprising:

detecting, in the pipelined microprocessor, a second instruction using second base and offset address values to store data into a second memory location that was previously read from, wherein the second instruction is detected based upon the second base and offset address values and without computing a memory address equaling the second base address value added to the offset address values in detecting the second instruction.

14.-15. (Canceled).

16. (Previously Presented) A method for operating a pipelined microprocessor as claimed in claim 12, further comprising:

examining, in the pipelined microprocessor, base and offset address values used to access memory locations by store instructions that store data into the memory locations; and

detecting load instructions that load data from memory locations corresponding to base and offset address values identical to the base and offset address values used by the store instructions.

17. (Previously Presented) A method for operating a pipelined microprocessor as claimed in claim 13, further comprising:

examining, in the pipelined microprocessor, base and offset address values used to access memory locations by load instructions that load data from memory locations; and

detecting said instructions that store data into memory locations corresponding to base and offset address values identical to the base and offset address values used by the load instructions.

18. (Previously Presented) A method for operating a pipelined microprocessor as claimed in claim 16, further comprising:

detecting, in an instruction decode stage of the pipelined microprocessor, load instructions that load data from memory locations corresponding to offset address values and base address values identical to offset address values and base address values used by prior store instructions that store data into the memory locations; and

sending a bypass signal from a bypass element to an instruction execution stage of the pipelined microprocessor, wherein the bypass signal indicates that a load instruction uses a base address value and an offset address value identical to a base address value and an offset address value used by a prior store instruction.

19. (Previously Presented) A method for operating a pipelined microprocessor as claimed in claim 17, further comprising:

detecting, in an instruction decode stage of the pipelined microprocessor, store instructions that store data into memory locations using offset address values and base address values identical to offset address values and base address values used by prior load instructions that load data from memory locations; and

sending a bypass signal from a bypass element to an instruction execution stage of the pipelined microprocessor, wherein the bypass signal indicates that a load instruction uses a base address value and an offset address value identical to a base address value and an offset address value used by a prior store instruction.

20. (Currently Amended) A method for operating a pipelined microprocessor, comprising:

detecting a first instruction that stores data to a first memory location, the first instruction comprising syntax for computing an effective address for the first memory location;

detecting a second instruction that loads data from a second memory location, the second instruction comprising syntax for computing an effective address for said second memory location;

determining the syntax for the first instruction and the syntax for the second instruction;

using the syntax for the first instruction and the syntax for the second instruction to determine a relationship between the first memory location and the second memory location, without using the effective address of the first memory location or the effective address of the second memory location to determine the relationship between the first memory location [[nd]] and the second memory location; and

using the relationship to determine whether to perform one of the first instruction and the second instruction.

21. (Previously Presented) A method for operating a pipelined microprocessor as claimed in claim 20, wherein the syntax for the first instruction and the syntax for the second instruction refer to an identical memory location.